

Counting Sort

Introduction

Counting sort assumes that each of the n input elements is an integer in the range 0 to k , for some integer k . When $k = O(n)$, the sort runs in $\Theta(n)$ time. Time Complexity is $\Theta(n+k)$, if $k \ll n$, then complexity is $\Theta(n)$.

Example:

$A = \{2, 5, 3, 0, 2, 3, 0, 3\}$, here, $n = 8$, k varies from 0 to 5, hence $\text{mod}(k) < n$.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 2 | 5 | 3 | 0 | 2 | 3 | 0 | 3 |
|---|---|---|---|---|---|---|---|

COUNTING-SORT(A, B, k)

//let C [0-k] be a new array (Count array)

for i = 0 to k

 C[i]= 0

for j =1 to A.length

 C[A[j]]= C[A[j]]+1

// C[i] now contains the number of elements equal to i.

for i = 1 to k

 C[i]= C[i] + C[i-1]

for j =A.length down to 1

 B[C[A[j]]]=A[j]

 C[A[j]] = C[A[j]]-1